Sub A1

Claim 1. A method for executing a target application on a host processor comprising the steps of:

translating into host instructions each of a sequence of target instructions,

storing the translated host instructions,

executing the stored host instructions, and

responding to an exception during execution of a stored translated instruction by rolling back to a point in execution at which correct state of a target processor is known, and

interpreting each target instruction in order from the point in execution at which correct state of a target processor is known.

Claim 2. A method as claimed in Claim 1 which further comprises:

collecting statistics regarding the execution of sequences of instructions which are interpreted.

Claim 3 . A method for executing a target application on a host processor comprising the steps of:

executing host instructions representing each target instruction of the target application;

responding to an exception during execution of host instructions representing a target instruction by returning to a point in execution of the target application at which correct state of a target processor is known; and

Cont A1

thereafter executing host instructions by interpretation of the target instruction until the point of the exception.

Claim 4. A method as claimed in Claim 3 which further comprises collecting statistics regarding the execution of sequences of target instructions which are executed.

Claim 5. A method as claimed in Claim 4 in which the statistics include the number of times the sequence of target instructions have executed.

Claim 6. A method as claimed in Claim 4 in which the statistics include address of an instruction to which a target instruction including a branch operation branches.

Claim 7. A method as claimed in Claim 4 in which the statistics include a likelihood of a branch being taken.

Claim 8. A system for executing a target application designed for execution on a target processor on a host processor having an instruction set different than that of the target processor comprising:

means for translating sequences of target instructions and storing each translated sequence of instructions,

means for selecting a stored translated sequence of instructions for execution,

means for responding to an exception during execution of a stored translated instruction by rolling back to a point in execution at which correct state of a target processor is known, and

Cont A1

means for interpreting each target instruction in order from a point in execution at which correct state of a target processor is known through the target instruction causing the exception.

Claim 9. A system as claimed in Claim 8 in which the means for interpreting is an interpreter software executing on the host processor, and

the means for translating is dynamic translation software executing on the host processor.

Claim 10. A method for executing a target application on a host processor comprising the steps of:

interpreting each of a sequence of target instructions a plurality of times,

collecting statistics regarding the execution of the instructions in the sequence of instructions as the sequence is interpreted,

using the statistics to determine when to cease interpreting target instructions,

translating into host instructions each of the sequence of target instructions after stopping interpreting the sequence,

storing the translated host instructions, and

executing the stored host instructions when the sequence of target instructions is to be executed.

Claim 11. A method as claimed in Claim 10 in which the statistics include the number of times the sequence of target instructions have executed.

Claim 12. A method as claimed in Claim 10 in which the statistics include an address of an instruction to which a target instruction including a branch operation branches.

Claim 13. A method as claimed in Claim 10 in which the statistics include a likelihood of a branch being taken.

Add A1